REMARKS

Claims 1-5 and 7-45 are pending in the application.

Claims 1-5 and 7-45 stand rejected.

Claim Interpretation

Applicants respectfully disagree with the assertion that "the 'ASCII string' as claimed refers to a method of encoding coefficients for a polynomial that is being programmed using a hardware description language and that any 'string' of values used for such a purpose is functionally equivalent to an 'ASCII string'." Final Office Action, p. 3. Applicant again notes that "[a]ll words in a claim must be considered in judging the patentability of that claim against the prior art." *In re Wilson*, 424 F.2d 1382, 165 USPQ 494, 496 (C.C.P.A. 1970). Accordingly, it is improper to ignore the term "ASCII" when interpreting the claims.

Applicant also disputes the inherency assertion on page 3 of the Final Office Action: "[T]he Examiner notes that in the *Foxcraft* [sic] reference it is *inherent* that there be a storage of "*strings*" coeffecients in order to properly perform a "*Galois Field Multiplier*" as claimed and enabled by Applicant's claim language and specification... the Examiner respectfully asserts that in order to enable the claimed limitations of [Applicant's claim 1], requires that some method of storing the polynomials coefficients" (emphasis in original). The Examiner's reasoning appears to be that Foxcroft must inherently include a particular feature recited in the Applicant's claims in order to properly implement the invention that the Applicant has claimed and enabled. Applicant notes that, "[i]n relying on a theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art." *Ex parte Levy*, 17 U.S.P.Q.2d 1461, 1464 (Bd. Pat. App. & Inter. 1990) (emphasis in original). Accordingly, the proper standard for inherency is whether the inherent characteristic flows from the teachings of the prior art, not whether the allegedly inherent characteristic flows from the teachings of Applicant's specification. Since no explanation as to why the allegedly inherent characteristic flows from the teachings of the prior art has been provided, Applicant respectfully requests the withdrawal of this assertion.

Rejection of Claims under 35 U.S.C. § 103

Claim 7 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Foxcroft, U.S. Patent No. 5,818,855 (hereinafter referred to as "Foxcroft"), in view of Ko, et al., U.S. Patent No. 5,905,664 (hereinafter referred to as "Ko"). The cited art fails to teach or suggest "producing one or more parallel equations in a hardware description language, wherein the producing comprises simulating a serial circuit; and merging the one or more parallel equations into a hardware description language implementation of a Galois Field circuit," as recited in claim 7. Accordingly, Applicant respectfully traverses this rejection.

The Examiner cites cols. 8-10 of Foxcroft as teaching "producing one or more parallel equations in a hardware description language." Final Office Action, p. 5. This portion of Foxcroft shows how two Galois Field multipliers (a conventional general Galois Field multiplier and Galois Field multiplier 116 of FIG. 10 of Foxcroft) are implemented using a standard hardware description language that conforms to IEEE Standard 1364-1995. Foxcroft, col. 8, lines 23-27. Foxcroft provides this example in order to show how the area reports for the two Galois Field multipliers differ from each other. Foxcroft, col. 9, line 34 - col. 10, line 55. Applicant notes that the mere presence of equations in a hardware description language neither teaches nor suggests a particular technique for producing one or more parallel equations. In other words, while Foxcroft does show equations, Foxcroft provides no information about how these equations were produced. Accordingly, Foxcroft clearly does not teach or suggest "producing one or more parallel equations in a hardware description language." For at least this reason, claim 7 is patentable over the cited art.

As noted on page 6 of the Final Office Action, "the [Foxcroft] reference does not expressly disclose the simulation of a serial circuit used for complex polynomials." Furthermore, Foxcroft provides no suggestion that "the producing [the one or more parallel equations] comprises simulating a serial circuit." As noted above, Foxcroft simply states that these equations are used to describe a conventional Galois field multiplier and Galois Field multiplier of FIG. 10. Foxcroft, cols. 8-9. No description of the techniques used to generate these equations, or suggestion relating to such techniques, is provided in the cited portions of Foxcroft.

Ko is cited as teaching "simulating a serial circuit for complex polynomials." Final Office Action, p. 10. In particular, the Final Office Action cites FIGs. 4 and 5, which show methods used to verify the circuit of FIG. 1, of Ko, and portions of col. 5. "FIG. 1 schematically illustrates a circuit 100 for receiving the bits of a code vector having coefficients a_{39} through a_8 and determining the remainder of dividing the code vector by a degree 8 generator polynomial." Ko, col. 2, lines 33-43. FIG. 4 "is a flowchart that illustrates a process for verifying the FIG. 1 circuit," while FIG. 5 "illustrates a simulation for use in the [verification process] of FIG. 4." Ko, col. 2 , lines 26-30. The circuit of FIG. 1 is designed according to the techniques described in claim 1 and cols. 3 and 4 of Ko. More detail regarding the verification process is provided in col. 5:

> In general, the verification for observable behaviors can be done by simulation or formal methods. For those non-observable behaviors, they can only be verified by formal methods...
>
> The formal verification model generation procedure produces CTL models describing the desired functionality for the parallel modulo, the corresponding serial modulo and relationship between the two. The basic concepts about the verification model is to relate the behavior of the parallel and the serial modulo (LFSR) in time.
>
> For the example of describes a 39-degree polynomial modulo by a 7-degree polynomial, the CTL model states that for all computation paths, all the outputs of the parallel modulo at the 0th cycle are equivalent to that of the serial modulo at the 32nd cycle. The RTL generation and formal verification process may be automated as shown in FIG. 4. The automated methodology consists of three steps: 1) the designer provides two parameters which are the dividend and divisor polynomials, 2) the model generator generates an RTL parallel model in Verilog or VHDL hardware description language and an associated formal self-verification model in which the desired functionality is stated, and 3) the formal self-verification model is fed into the SMV formal verification frame work to check if the generated parallel modulo preserves the desired properties (functionality). IF the verification failed, the verification framework produces counter vectors indicating where the parallel model is incorrect." Ko, col. 5, lines 7-37.

As this shows, Ko teaches using a serial model (which performs serial modulo operation) and a parallel model (which performs a parallel modulo operation and should generate the same result as the serial model, but in fewer cycles) in order to verify that the parallel model is operating properly. While Ko does disclose using simulation, the simulation described in Ko is performed in the context of verification, not in the context of producing one or more parallel equations. Thus, Ko does not disclose the type of simulation that is recited in claim 7.

Furthermore, there is no suggestion, or reason to expect such a suggestion, to use the simulation described in Ko's verification process to produce one or more parallel equations in a hardware description language. Ko already teaches how the parallel circuit is produced, without the use of simulation, and no simulation is performed until a parallel circuit has already been produced. In particular, Ko's process for producing the parallel circuit described by the parallel model is set forth in claim 1 and cols. 3 and 4 of Ko, and clearly does not involve or suggest simulating a serial circuit to produce parallel equations. Additionally, any parallel equations implemented by the parallel model in Ko have already been produced when the verification process that uses the serial model takes place (otherwise, there would be no parallel model to verify), making it unnecessary to produce any equations during verification. Ko clearly does not teach or suggest simulate a serial circuit as part of the process of producing one or more parallel equations.

Thus, Ko and Foxcroft, both alone and in combination, clearly do not teach or suggest producing parallel equations, where the production of the parallel equations involves simulating a serial circuit. At best, the combination of Foxcroft and Ko teaches (1) simulating a circuit by using several equations to describe the circuit in a hardware description language and (2) verifying a parallel circuit by comparing the operation of both parallel model of the parallel circuit and a serial model designed to provide the same functionality as the parallel circuit. This combination clearly neither teaches nor suggests "producing one or more parallel equations in a hardware description language, wherein the producing comprises simulating a serial circuit." Accordingly, the cited combination fails to teach or suggest claim 7.

Furthermore, there is no suggestion to combine the references. The stated reason for the combination is that, "by using the *Ko et al.* method, the remainders of modulo 2 polynomial division can be achieved in just one clock cycle." While this assertion does state an advantage of the circuit disclosed in Ko, there is no suggestion that it would be desirable to include such a circuit in the Galois field multiplier taught in Foxcroft. Regardless of how well the circuit of Ko performs, there does not appear to be any need to obtain the remainders of modulo 2 polynomial division in the system taught in Foxcroft, which is concerned with performing Galois field multiplication. Adding additional, unused circuitry would appear to simply increase the expense and complexity of Foxcroft's system, without providing any benefit. No other explanation of how and why the teachings of Ko would be applicable and desirable in the context of Foxcroft's

system have been provided. Since no suggestion to combine the references has been provided, Applicant respectfully requests the withdrawal of this rejection.

Claims 26 and 36 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Tomizawa et al., U.S. Patent No. 5,574,717 (hereinafter referred to as "Tomizawa"), in view of Ko. The cited art fails to teach or suggest an "FEC decoder [that] comprises a first circuit that implements one or more parallel equations, [where] the one or more parallel equations are generated by simulating a serial circuit," as recited in claim 26. Applicant respectfully traverses this rejection.

As noted on page 6 of the Final Office Action, Tomizawa does not disclose simulating a serial circuit. Furthermore, Tomizawa neither teaches nor suggests one or more parallel equations that are generated by simulating a serial circuit (no art has been cited as teaching this feature of claims 26 and 36). The Final Office Action relies on Ko to teach "simulating a serial circuit for complex polynomials." However, as explained above, Ko simply teaches a technique for verifying a parallel model using a serial model, in a context that is completely separate from the context of generating parallel equations. Neither Tomizawa nor Ko teaches or suggests using simulation to generate parallel equations. Accordingly, the cited art does not teach or suggest a process in which "one or more parallel equations are generated by simulating a serial circuit," and thus the combination of references fails to teach or suggest claim 26. Claim 36 is patentable over the cited art for similar reasons.

Claims 1, 2-5, 8-15, 16, 17, 18, 19, and 20-25 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Foxcroft, in view of Ko, and in further view of Oskouy et al., U.S. Patent No. 5,673,279 (hereinafter referred to as "Oskouy") and in further view of Kao, U.S. Patent No. 5,642,367 (hereinafter referred to as "Kao"). Applicant respectfully traverses this rejection.

Claim 1 recites:

> implementing a serial circuit representing the complex polynomial equation in a software program; wherein implementing the serial circuit includes
> storing a plurality of ASCII strings in each of a plurality of storage elements, wherein the plurality of ASCII strings represent a plurality of initial values of the serial circuit,

storing one or more ASCII strings in each of one or more data structures wherein the one or more ASCII strings represent one or more mathematical operations in the serial circuit, and
simulating the serial circuit to produce a plurality of parallel equations, wherein simulating the serial circuit includes
simulating the serial circuit for a plurality of cycles as required to produce one output represented by the plurality of parallel equations. (emphasis added)

The cited art fails to teach or suggest "simulating the serial circuit to produce a plurality of parallel equations." As set forth with respect to claim 7 above, the combination of Foxcroft and Ko does not teach or suggest a feature like this. Additionally, the combination of Foxcroft, Ko, Oskouy, and Kao fails to teach or suggest this feature.

As noted above, Foxcroft does not teach simulating a serial circuit, and neither teaches nor suggests performing such simulation in order to produce a plurality of parallel equations. The Final Office Action cites Oskouy and Ko as disclosing simulation of a serial device and serial circuit respectively; however, the Final Office Action has not cited any art as teaching "simulating the serial circuit to produce a plurality of parallel equations," as recited in claim 1.

As noted above, while Ko does disclose simulating a serial model as part of the process of verifying a parallel model, Ko does not teach or suggest simulating a serial circuit in order to generate parallel equations. Additionally, Applicant again asserts that there is no suggestion to combine Ko with Foxcroft, as explained in more detail above.

Oskouy describes simulating a network transporter (e.g., see Oskouy, Abstract). Oskouy verifies the network transporter using a simulated physical environment. However, the simulation techniques taught in Oskouy are used to test a network transporter, not a serial circuit that is implemented in a software program by, among other things, storing ASCII strings of the type recited in claim 1. Thus, Oskouy is simulating a different type of circuit than is recited in claim 1. Furthermore, the result of the simulation taught in Oskouy is that the proper operation of a network transporter can be verified, not the production of one or more parallel equations. Thus, while Oskouy does teach simulating a circuit, Oskouy does not teach simulating a serial circuit to produce a plurality of parallel equations, as recited in claim 1. Oskouy, both alone and in combination with the other references, does not teach or suggest simulating a serial circuit in the context of producing one or more parallel equations.

The Final Office Action also cites Kao as disclosing that, "in order to properly simulate a forward error correcting circuit design that the circuit performing the calculations of a Galois

field must operate for a plurality of cycles." Final Office Action, p. 8. Again, while Kao does disclose a circuit that can be simulated, the cited portions of Kao neither teach nor suggest generating parallel equations by simulating a serial circuit.

At best, the combination of Foxcroft, Oskouy, Kao, and Ko teaches (1) using equations to describe a circuit in a hardware description language and then simulating the circuit, (2) simulating a network transporter, (3) operating a forward error correcting circuit for multiple cycles, and (4) verifying a parallel model using a serial model. In all of these references, any circuits being simulated are simulated for purposes of verifying the operation of the circuit, not for purposes of generating one or more parallel equations. Accordingly, the cited portions of the references, both alone and in combination, clearly fail to teach or suggest "simulating the serial circuit to produce a plurality of parallel equations," as recited in claim 1. Thus, claim 1 and dependent claims 2-5 are patentable over the cited art for at least the foregoing reasons. Claims 8-15, 16, 17, 18, 19, and 20-25 are patentable over the cited art for similar reasons.

Claims 27-30, 32, 34, 35, 37-40, 42, 44 and 45 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Tomizawa in view of Ko and in further view of Foxcroft. Claims 33 and 43 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Tomizawa in view of Foxcroft and in view of Ko and in further view of Oskouy. Claims 31 and 41 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Tomizawa in view of Foxcroft and in view of Ko and in further view of Kao. These claims are patentable over the cited art for reasons similar to those provided above.

CONCLUSION

In view of the amendments and remarks set forth herein, the application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the Examiner is invited to telephone the undersigned at 512-439-5087.

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on **May 16, 2005**

Brenna A Brock 5.16.2005
Attorney for Applicants Date of Signature

Respectfully submitted,

Brenna A Brock

Brenna A. Brock
Attorney for Applicants
Reg. No. 48,509
Telephone: (512) 439-5087
Facsimile: (512) 439-5099